Remarks

Claims 1-7, 10-12, 15, 17-23, 26-28, 31 and 47-48 are pending. All pending claims stand rejected under 35 USC §102. Claims 1, 10, 15, 17, 26, and 31 have been amended.

Claim Rejections – 35 USC §102: The Examiner rejected Claim 1-7, 10-12, 15, 17-23, 26-28, 31 and 47-48 under §102(e) as being anticipated by Brown (US Pub. No. 2002/0174180). A claim is anticipated only if the cited reference teaches or suggests the combination of elements required by that claim. As explained below, Brown fails to teach or suggest one or more elements required by each of the pending claims. The rejections are, therefore, improper.

Claim 1 is directed to a coordinated push synchronization method and includes the following combination of elements.

1. detecting changes to a local application data store;
2. identifying a record affected by a detected change;
3. pushing the identified record to a remote application data store;
4. ascertaining whether the pushed record, in its current form as affected by the detected change, has already been replicated or deleted in the remote application data store in order to determine whether the remote application data store will be updated with the pushed record;
5. if not, updating the remote application data store with the pushed record and identifying the pushed record in the remote application data store as having been pushed from the local application data store to the remote application data store, otherwise ignoring the pushed record.

With respect to the fourth element listed above, Brown does not teach or suggest ascertaining whether a record, ***pushed to a remote application data store,*** in its current form has already been replicated or deleted in the remote application data

11

store *in order to determine whether the remote application data store will be updated with the pushed record*. Furthermore, with respect to the fifth element listed above Brown does not teach or suggest that where it ascertained that the pushed record has not been replicated or deleted, the remote application data store is updated with the pushed record and *the pushed record is identified within the remote application store as having been pushed from the local application data store to the remote application data store.*

The Examiner mistakenly contends that Brown paragraphs 80-83 teach the fourth limitation and that paragraph 66 teaches the fifth limitation listed above. Brown is directed to a client server model for synchronizing files. Brown, Title and Abstract. The first passage relied upon by the Examiner with respect to the fourth limitation are reproduced as follows:

[0080] SID is in the SSD but not in the CSD. The SID identifies a new file or directory that exists on the server and needs to be replicated on the client. In the case of a file it must be downloaded; directories just need to be created. In this case, the SSD also includes the metadata item for the file or directory.

[0081] SID is in the CSD but not in the SSD. The SID identifies a file or directory that has been deleted on the server but is still present on the client. The client SA must delete the file or directory.

[0082] SID is present in both sets but in different directories. The SID identifies a file or directory that has been moved from one directory to another on the server. The client SA must replicate the move. In this case, the SSD also includes the metadata item for the file or directory that includes the name of the file or directory. The name must be checked in case the file was also renamed on the server.

[0083] SID is present in both sets in the same directory. The SID identifies a file that has not been moved or deleted on the server. The client SA must still check the SSD for a metadata item with the SID in case the file or directory was renamed on the server.

Brown, paragraphs [0080] through [0083].

12

Paragraphs 80-83 describe situations resulting from the comparison of server synchronization data (SSD) with client synchronization data. Brown, para. [0079]. Server synchronization data contains the server's current server synch index (SSI), Server identifiers (SIDs) of the directories with changes, server identifiers (SIDs) of the child directory and child file items for each changed directory, and server metadata items for any items with file synch indexes (FSIs) greater than the client synch Index (CSI) passed in by the client's sync poll call. Brown, paragraphs [0071]-[0075]. Metadata can include a filename, client and server indexes, and a file synch index. Brown, para. [0066]. Client synchronization data includes server identifiers (SIDs) of all the server directories that existed for the previous client server index (CSI) and, for each directory server identifier (SID), the set of directory and file server identifiers (SIDs) contained in the directory that existed for the previous client server Index (CSI). Brown, paragraphs [0076]-[0078].

The examiner cited Brown, paragraphs [0080] through [0083] asserting that the passage teaches ascertaining whether a record, pushed to a remote application data store, in its current form has already been replicated or deleted in the remote application data store. Contrary to the Examiner's contention, Brown mentions nothing of ascertaining whether a remote application data store has been updated with a record that has been pushed to that remote application data store in order to determine whether the remote data store will be updated with the pushed record.

The passage relied upon by the Examiner merely describes different actions taken depending upon an ascertained condition. In each case a data store will be updated. In paragraph [0080], a condition indicates a new file or directory that exists on a server that needs to be replicated on a client. Since the file or directory has not been pushed to the client, no determination can be made as to whether such a pushed record has already been replicated on the client through a previous update of the client with the pushed file or directory. Logically, one cannot ascertain a status of a pushed record if that record has not been pushed.

13

In paragraph [0081], the condition indicates that a file or directory has been deleted on a server but is still present on a client and that the client must delete the file or directory. Since a record indicating that the file or directory has been deleted has not been pushed to the client, no determination can be made as to whether such a pushed record has already been replicated on the client through a previous deletion of the corresponding file or directory. Logically, one cannot ascertain a status of a pushed record if that record has not been pushed.

In paragraph [0082], the condition indicates that file or directory that has been moved from one directory to another on the server and that the client must replicate the move. Since a record indicating the move of the file or directory has been not been pushed to the client, no determination can be made as to whether such a pushed record has already been replicated on the client through a previous move of the corresponding file or directory. Logically, one cannot ascertain a status of a pushed record if that record has not been pushed.

In paragraph [0083], the condition indicates that a file or directory may have been renamed on the server. Since a record indicating the renaming of the file or directory has been not been pushed to the client, no determination can be made as to whether such a pushed record has already been replicated on the client through a previous renaming of the corresponding file or directory. Logically, one cannot ascertain a status of a pushed record if that record has not been pushed.

None of the paragraphs describe a situation in which a record or other data is pushed from a server to a client followed by a determination as to whether the pushed record, in its current form as affected by a detected change, has already been replicated or deleted on that client in order to determine whether the client will be updated with the pushed record.

Furthermore, with respect to the fifth element of Claim 1, Brown fails to teach or suggest conditionally updating the remote application data store with the pushed

14

record and identifying the pushed record in the remote application data store as having been pushed from the local application data store to the remote application data store. In other words, Brown does not teach or suggest ignoring the pushed record if it is ascertained that the pushed record has not already been replicated in the remote application data store.

The Examiner contends that Brown, paragraph [0066] teaches this fifth limitation of Claim 1. That passage of Brown is reproduced as follows:

> [0066] Metadata 330 shows the metadata for file 320 as stored within CSFS 335, part of client SA 337. (Although metadata are not shown for the other files and folders within folder 302, a person skilled in the art will recognize that such metadata exist.) In metadata 330, file 320 is shown as having a name (which is typically not encrypted, although the name can be encrypted in an alternative embodiment of the invention), a CID of 0.times.62, a SID of 0.times.2A, a FSI of 35, the change time of the file, and the flags used in the synchronization process (such as identifying metadata *items that need to be pushed to the server*). Note that metadata 330 is not shown to store the data of file 320, which is stored in the native operating system of computer 130 within the folder structure, as expected.

Brown, paragraph [0066] (emphasis added).

The passage mentions nothing of conditionally updating a server with a pushed file or directory after it is ascertained that the file or directory pushed to that server has already not been replicated on that server. The passage mentions nothing of ignoring a pushed file or directory after it is ascertained that the file or directory pushed to that server has already been replicated on that server. Moreover, the passage mentions nothing of identifying a file or directory pushed to the server as having been pushed from the client to the server. Paragraph [0066] merely discusses metadata stored on a client for a file or folder on that client. That metadata includes flags that indicate whether a file or folder on that client <u>needs</u> to be pushed to a server. Therefore, the client's metadata flags <u>do not</u> and <u>cannot</u> indicate that a file or directory has been pushed. The flags indicate whether it needs

15

to be pushed. Logically, a file or directory that needs to be pushed cannot have already been pushed.

For at least these reasons, Claim 1 is patentable over Brown. Claims 2-4 and 47 are thus also felt to distinguish over Brown based on their dependency from Claim 1.

Claim 5 is directed to a coordinated user-initiated synchronization method and includes the following combination of elements:

1. detecting changes to a local application data store;
2. identifying a record affected by a detected change;
3. ascertaining whether the identified record, in its current form as affected by the detected change, was pushed to the local application data store from a remote application data store; and
4. if not, synchronizing the remote application data store with the local application data store.

The Examiner contends that Brown paragraph 40 teaches the third and fourth elements listed above. Paragraph 40 is reproduced as follows:

> The synchronization process is initiated by a client SA when it makes a sync poll call to the server, passing it the client sync index (CSI) identifying its current known state of the account. If the SSI value matches the CSI value passed in by the polling client, the client is up to date with the current state of its server account. In that case, the server SFS returns the SSI (having the same value as the CSI) back to the client. Otherwise, the server SFS returns the new higher SSI along with the server metadata information the client needs to transition its account from its current state to the server's current state.

Brown, paragraph [0040].

Paragraph [0040] merely describes a client application(SA) that passes a client sync index (CSI) to a server. If the CSI matches server sync index (SSI) then the client and server are already synchronized. If the server determines that the CSI

16

and SSI do not match, the server returns the SSI and metadata information so that the client's state can be updated to match that of the server.

Nothing in paragraph 40 mentions or suggests ascertaining whether a record (a file or directory) of a client that that has been identified as being affected by a detected change on that client was in fact pushed to the client from a server in its current changed form. Furthermore Brown fails to teach or suggest conditionally synchronizing the client with the server if it is ascertained that the identified record in its current changed form was not pushed to the client from the server.

For at least these reasons, Claim 5 is felt to distinguish over Brown. Claims 6, 7, and 48 are thus also felt to distinguish over Brown based on their dependency from Claim 5.

Claim 10 is directed to a coordinated push and user-initiated synchronization method and includes the following combination of elements.

1. detecting changes to a local application data store;
2. identifying a first record in the local application data store affected by a detected change;
3. pushing the first record to a remote application data store;
4. ascertaining whether the pushed record, in its current form as affected by the detected change, has already been replicated in or deleted from the remote application data store and, if not, updating the remote application data store with the pushed record;
5. detecting changes to the remote application data store;
6. identifying a second record in the remote application data store affected by a detected change;
7. ascertaining whether the second record, in its current form as affected by the detected change, has already been pushed into the remote application data store in order to determine whether the remote

17

application data store will be updated with the pushed record and, if
not, synchronizing the remote application data store with the local
application data store, otherwise ignoring the pushed record.

As with Claim 5, the Examiner contends that Brown paragraph 40 teaches the fourth
element listed above. As with Claim 1, the Examiner contends that Brown
paragraphs 79-83 teach the seventh element listed above. As made clear above
with reference to Claims 1 and 5, The Examiner's contentions are mistaken. Brown
fails to teach or suggest the fourth and seventh elements.

For at least these reasons, Claim 10 is felt to distinguish over Brown. Claims
11, 12, and 15 are thus also felt to distinguish over Brown based on their
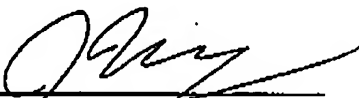dependency from Claim 10.

Claim 17 is directed to a computer readable medium having instructions for
performing the method steps of Claim 1. For the same reasons Claim 1
distinguishes over Brown so does Claim 17. Claims 18-20 are thus also felt to
distinguish over Brown based on their dependency from Claim 17.

Claim 21 is directed to a computer readable medium having instructions for
performing the method steps of Claim 5. For the same reasons Claim 5
distinguishes over Brown so does Claim 21. Claims 22 and 23 are thus also felt to
distinguish over Brown based on their dependency from Claim 21.

Claim 26 is directed to a computer readable medium having instructions for
performing the method steps of Claim 10. For the same reasons Claim 10
distinguishes over Brown so does Claim 26. Claims 27, 28, and 31 are thus also felt
to distinguish over Brown based on their dependency from Claim 26.

18

Conclusion: All pending claims are felt to be in condition for allowance. The foregoing is believed to be a complete response to the outstanding office action.

Respectfully submitted,
Richard Detweiler, et al

By _____
Jack H. McKinney
Reg. No. 45,685

July 27, 2005

19